

This listing of claims will replace all prior versions, and listings, of claims in the application.

**Listing of Claims:**

1. (Currently Amended) A method for modeling a web server, comprising:
  - identifying a plurality of sub-systems for the server, said plurality of sub-systems comprising a transaction control protocol/internet protocol (TCP/IP) sub-system, a hypertext transfer protocol (HTTP) sub-system, and an input/output (I/O) sub-system;
    - representing each sub-system as a queue, with each queue operably coupled together, wherein said TCP/IP sub-system comprises a first finite listen queue served by a listener daemon, said HTTP sub-system comprises a second finite listen queue served by one or more multi-threaded HTTP daemons with  $N_{http}$  separate server threads, and said I/O sub-system comprises a finite number  $N_{buf}$  of network buffers served by an input/output controller; and
      - iteratively adjusting an arrival rate and a service time for each queue to account for performance by other queues.
2. (Currently Amended) The method of claim 1, [wherein said plurality of sub-systems comprises one or more of a set comprising a transaction control protocol/internet protocol sub-system, a hypertext transfer protocol sub-system, an input/output sub-system, and] further comprising an active script component sub-system.
3. (Cancelled)
4. (Cancelled)
5. (Cancelled)
6. (Cancelled)

7. (Currently Amended) The method of claim [6] 1, wherein said input/output controller serves each network buffer using a polling system.

8. (Currently Amended) The method of claim [2] 1, wherein said transaction control protocol/internet protocol sub-system TCP/IP is represented as an  $M(\lambda_{file}) / M(\tau_{tcp}) / N_{tcp} / 0$  blocking system.

9. (Currently Amended) The method of claim [2] 1, wherein said hypertext transfer protocol sub-system is represented as an  $M(\lambda_{http}) / M(\tau_{http}) / N_{http} / Q_{http}$  queu[e]ing system.

10. (Currently Amended) The method of claim [2] 1, wherein said input/output sub-system is represented as an  $M(\lambda_{buf}) / M(\tau_{buf}) / N_{buf} / \infty$  queu[e]ing system.

11. (Currently Amended) A method for modeling a web server, comprising:

(a) identifying for the server a transaction control protocol/internet protocol (TCP/IP) sub-system, a hypertext transfer protocol (HTTP) subsystem, and an input/output (I/O) sub-system;

(b) representing each sub-system as a queuing system, wherein said TCP/IP sub-system comprises a first finite listen queue served by a listener daemon, said HTTP sub-system comprises a second finite listen queue served by one or more multi-threaded HTTP daemons with  $N_{http}$  separate server threads, and said I/O sub-system comprises a finite number  $N_{buf}$  of network buffers served by an input/output controller;

(c) computing an upper bound performance for said I/O sub-system by assuming a first predetermined blocking value for said TCP/IP sub-system and HTTP sub-system;

(d) computing an upper bound performance for said TCP/IP sub-system and HTIP sub-system by assuming a first predetermined I/O sub-system waiting time;

(e) computing a lower bound I/O performance by assuming a second

predetermined blocking value for said TCP/IP sub-system and HTTP sub-system;

(f) computing a lower bound performance for said TCP/IP sub-system and HTTP sub-system by assuming a second predetermined I/O sub-system waiting time; and

(g) repeating steps (c) - (f) to generate successively tighter bounds until convergence.

12. (Currently Amended) A machine-readable medium whose contents cause a computer system to model a web server, by performing the steps of:

identifying a plurality of sub-systems for the server, said plurality of sub-systems comprising a transaction control protocol/internet protocol (TCP/IP) sub-system, a hypertext transfer protocol (HTTP) sub-system, and an input/output (I/O) sub-system;

representing each sub-system as a queue, with each queue operably coupled together, wherein said TCP/IP sub-system comprises a first finite listen queue served by a listener daemon, said HTTP sub-system comprises a second finite listen queue served by one or more multi-threaded HTTP daemons with  $N_{http}$  separate server threads, and said I/O sub-system comprises a finite number  $N_{buf}$  of network buffers served by an input/output controller; and

iteratively adjusting an arrival rate and a service time for each queue to account for performance by other queues.

13. (Currently Amended) The machine-readable medium of claim 12, [wherein said plurality of sub-systems comprises one or more of a set comprising a transaction control protocol/internet protocol sub-system, a hypertext transfer protocol sub-system, an input/output sub-system, and] further comprising an active script component sub-system.

14. (Cancelled)

15. (Cancelled)

16. (Cancelled)

17. (Cancelled)

18. (Currently Amended) The machine-readable medium of claim [17] 12, wherein said input/output controller serves each network buffer using a polling system.

19. (Currently Amended) The machine-readable medium of claim [13] 12, wherein said transaction control protocol/internet protocol sub-system TCP/IP is represented as an  $M(\lambda_{file}) / M(\tau_{tcp}) / N_{tcp} / 0$  blocking system.

20. (Currently Amended) The machine-readable medium of claim [13] 12, wherein said hypertext transfer protocol sub-system is represented as an  $M(\lambda_{http}) / M(\tau_{http}) / N_{http} / Q_{http}$  queuing system.

21. (Currently Amended) The machine-readable medium of claim [13] 12, wherein said input/output sub-system is represented as an  $M(\lambda_{buf}) / M(\tau_{buf}) / N_{buf} / \infty$  queuing system.

22. (Currently Amended) A machine-readable medium for modeling a web server, comprising:

(a) identifying for the server a transaction control protocol/internet protocol (TCP/IP) sub-system, a hypertext transfer protocol (HTTP) sub-system, and an input/output (I/O) sub-system;

(b) representing each sub-system as a queuing system, wherein said TCP/IP sub-system comprises a first finite listen queue served by a listener daemon, said HTTP sub-system comprises a second finite listen queue served by one or more multi-threaded HTTP daemons with  $N_{http}$  separate server threads, and said I/O sub-system comprises a finite number  $N_{buf}$  of network buffers served by an input/output controller;

- (c) computing an upper bound performance for said I/O sub-system by assuming a first predetermined blocking value for said TCP/IP sub-system and HTTP sub-system;
- (d) computing an upper bound performance for said TCP/IP sub-system and HTTP sub-system by assuming a first predetermined I/O sub-system waiting time;
- (e) computing a lower bound I/O performance by assuming a second predetermined blocking value for said TCP/IP sub-system and HTTP sub-system;
- (f) computing a lower bound performance for said TCP/IP sub-system and HTTP sub-system by assuming a second predetermined I/O sub-system waiting time; and
- (g) repeating steps (c) - (f) to generate successively tighter bounds until convergence.